


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used

built in and test and generation and regenerat and execute and result

 Found **96,287** of **145,831**

Sort results by


☒ Save results to a Binder

 Try an [Advanced Search](#)

Display results


☒ Search Tips

 Try this search in [The ACM Guide](#)
☐ Open results in a new window

Results 1 - 20 of 200

 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

 Relevance scale ☐ ☐ ☐ ☐ ☐

1 [BIFEST: a built-in intermediate fault effect sensing and test generation system for CMOS bridging faults](#)

Kuen-Jong Lee, Jing-Jou Tang, Tsung-Chu Huang

 April 1999 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 4 Issue 2

 Full text available: pdf(208.73 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper presents BIFEST, an ATPG system that employs the built-in intermediate voltage test technique in an efficient ATPG process to deal with CMOS bridging faults. Fast and accurate calculations of the intermediate bridging voltages and the variant threshold tolerance margins on a resistive bridging fault model are presented. A PODEM-like, PPSFP-based ATPG process is developed to generate test patterns for faults that are conventionally logic-testable. The remaining faults are then dea ...

2 [Fast detection of communication patterns in distributed executions](#)

Thomas Kunz, Michiel F. H. Seuren

 November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

 Full text available: pdf(4.21 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

3 [Automatic testing equivalence verification of spi calculus specifications](#)

Luca Durante, Riccardo Sisto, Adriano Valenzano

 April 2003 **ACM Transactions on Software Engineering and Methodology (TOSEM)**, Volume 12 Issue 2

 Full text available: pdf(829.73 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Testing equivalence is a powerful means for expressing the security properties of cryptographic protocols, but its formal verification is a difficult task because of the quantification over contexts on which it is based. Previous articles have provided insights into using theorem-proving for the verification of testing equivalence of spi calculus specifications. This article addresses the same verification problem, but uses a state



[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: ☒ The ACM Digital Library ☐ The Guide

SEARCH

THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction sur](#)

Terms used

built in and test and generation and regenerat and execute and result and device and under test and runtime

Sort results by

[Save results to a Binder](#)

[Try an Advanced Search](#)

Display results

[Search Tips](#)

[Try this search in The ACM Guide](#)

☐ Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐

1 [Fast detection of communication patterns in distributed executions](#)

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available: pdf(4.21 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

2 [Computing curricula 2001](#)

September 2001 **Journal on Educational Resources in Computing (JERIC)**

Full text available: pdf(613.63 KB) html(2.78 KB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

3 [Draft report on requirements for a common prototyping system](#)

R. P. Gabriel

March 1989 **ACM SIGPLAN Notices**, Volume 24 Issue 3

Full text available: pdf(4.76 MB)

Additional Information: [full citation](#), [citations](#), [index terms](#)

4 [The LRPD test: speculative run-time parallelization of loops with privatization and reduction parallelization](#)

Lawrence Rauchwerger, David Padua

June 1995 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1995 conference on Programming language design and implementation**, Volume 30 Issue 6

Full text available: pdf(1.74 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Current parallelizing compilers cannot identify a significant fraction of parallelizable loops because they have complex or statically insufficiently defined access patterns. As parallelizable loops arise frequently in practice, we advocate a novel framework for their identification: speculatively execut